

```

EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDDD FFFFFFFFFFFFFF
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDDD FFFFFFFFFFFFFF
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDDD FFFFFFFFFFFFFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEEEEEEEEEEEEEEEEE DDD DDD FFF
EEEEEEEEEEEEEEEEEE DDD DDD FFFFFFFF
EEEEEEEEEEEEEEEEEE DDD DDD FFFFFFFF
EEEEEEEEEEEEEEEEEE DDD DDD FFFFFFFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEE DDD DDD FFF
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDD FFF
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDD FFF
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDD FFF

```

[illegible]

ED

```

EEEEEEEEEE DDDDDDDD FFFFFFFF SSSSSSSS TTTTTTTTTT RRRRRRRR UU UU CCCCCCCC TTTTTTTTTT
EEEEEEEEEE DDDDDDDD FFFFFFFF SSSSSSSS TTTTTTTTTT RRRRRRRR UU UU CCCCCCCC TTTTTTTTTT
EE DD DD FF SS TT RR RR UU UU CC TT
EE DD DD FF SS TT RR RR UU UU CC TT
EE DD DD FF SS TT RR RR UU UU CC TT
EE DD DD FF SS TT RR RR UU UU CC TT
EEEEEEEEEE DD DD FFFFFFFF SSSSSS TT RRRRRRRR UU UU CC TT
EEEEEEEEEE DD DD FFFFFFFF SSSSSS TT RRRRRRRR UU UU CC TT
EE DD DD FF SS TT RR RR UU UU CC TT
EE DD DD FF SS TT RR RR UU UU CC TT
EE DD DD FF SS TT RR RR UU UU CC TT
EE DD DD FF SS TT RR RR UU UU CC TT
EEEEEEEEEE DDDDDDDD FF SSSSSSSS TT RR RRRRRRRR UU UU CC TT
EEEEEEEEEE DDDDDDDD FF SSSSSSSS TT RR RRRRRRRR UU UU CC TT
.....
.....
.....
.....

SSSSSSSS DDDDDDDD LL
SSSSSSSS DDDDDDDD LL
SS DD DD LL
SS DD DD LL
SS DD DD LL
SS DD DD LL
SSSSSS DD DD LL
SSSSSS DD DD LL
SS DD DD LL
SS DD DD LL
SS DD DD LL
SSSSSSSS DDDDDDDD LLLLLLLLLL
SSSSSSSS DDDDDDDD LLLLLLLLLL

```

```
MODULE      EDFSTRUCT;
/* EDFSTRUCT - Structure definitions for EDIT/FDL
/*
/*      Version      'V04-000'
/*
/******
/*
/*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
/*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
/*  ALL RIGHTS RESERVED.
/*
/*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
/*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
/*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
/*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
/*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
/*  TRANSFERRED.
/*
/*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
/*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
/*  CORPORATION.
/*
/*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
/*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
/*
/******

/+++
/*
/* FACILITY:  EDIT/FDL (EDF)
/*
/* ABSTRACT:
/*
/*      This file contains the SDL source for EDIT/FDL (EDF).
/*
/* ENVIRONMENT:
/*
/*      n/a
/*
/*--
/*
/* AUTHOR:  Ken Henderson      CREATION DATE:  Sep-1982
/*
/* MODIFIED BY:
/*
/*      V03-003 KFH0003      Ken Henderson      8 Sep 1983
/*      Fixed ".,." in constant list.
/*
/*      V03-002 KFH0002      Ken Henderson      26 Apr 1983
/*      Added offsets for ADD_KEY, DELETE_KEY
/*      scripts.
/*
/*      V03-001 KFH0001      Ken Henderson      14 Apr 1983
/*      Added offsets for SET_FUNCTION,
```

```
/*          GRANULARITY, PROMPTING, RESPONSES,
/*          ANALYSIS, SEGMENTED and OUTPUT.
/*

/** Non-QTAB offsets, for variables that don't correspond to questions.
/**
/** They should be ordered: IDATA, BDATA, RDATA, SDATA types.
/** (NON-STORAGE) means an identifier just generates a constant,
/** but will have an array location - which is never accessed.

/**
/** xxx_HIGH identifiers MUST immediately follow the xxx_LOW identifiers!!!
/**

/** IDATA OFFSETS FOLLOW

        #MARK0 = 0;

CONSTANT      (
              "FDL_FILL",
              "FINAL_DESIGN",          /** (NON-STORAGE)
              "FIRST_SCRIPT",          /** (NON-STORAGE)
              "WRITE_PLOT",
              "Y_HIGH",
              "Y_LOW",
              "Y_INCR",

              ) EQUALS #MARK0
              INCREMENT 1
              PREFIX "EDF$"
              COUNTER #MARK1;

/** BDATA OFFSETS FOLLOW

CONSTANT      (
              "DUMMY"

              ) EQUALS #MARK1+1
              INCREMENT 1
              PREFIX "EDF$"
              COUNTER #MARK2;

/** RDATA OFFSETS FOLLOW

CONSTANT      (
              "ADDED_FILL",
              "LOAD_FILL",

              ) EQUALS #MARK2+1
              INCREMENT 1
              PREFIX "EDF$"
              COUNTER #MARK3;
```


/** SDATA OFFSETS FOLLOW

```
CONSTANT      (  
    "PLACE HOLDER"  
    ) EQUALS #MARK3+1  
    INCREMENT 1  
    PREFIX "EDFS"  
    COUNTER #MARK4;
```

/** QTAB equivalent offsets that correspond to actual questions

/**

/** They should be ordered SDATA, RDATA, BDATA, IDATA types

/** SDATA OFFSETS FOLLOW

```
CONSTANT      (  
    "DATA_FILE_NAME",  
    "FDL_TITLE",  
    "KEY_NAME",  
    "ANALYSIS",  
    "OUTPUT"  
    ) EQUALS #MARK4+1  
    INCREMENT 1  
    PREFIX "EDFS"  
    COUNTER #MARK5;
```

/** RDATA OFFSETS FOLLOW

```
CONSTANT      (  
    "DATA_KEY_COMP",  
    "DATA_RECORD_COMP",  
    "INDEX_RECORD_COMP",  
    ) EQUALS #MARK5+1  
    INCREMENT 1  
    PREFIX "EDFS"  
    COUNTER #MARK6;
```

/** BDATA OFFSETS FOLLOW

```
CONSTANT      (  
    "KEY_COMP_WANTED",  
    "REC_COMP_WANTED",  
    "IDX_COMP_WANTED",  
    "ASCENDING_ADDED",  
    "ASCENDING_LOAD",  
    "BLOCK_SPAN",  
    "CONFIRM",  
    "SEGMENTED",  
    "GLOBAL_WANTED",  
    )
```

/** FUNCTION

```
"KEY_CHANGES",  
"KEY_DIST",  
"KEY_DUPS",  
"RETURN"
```

/** FUNCTION

```
) EQUALS #MARK6+1  
INCREMENT 1  
PREFIX 'EDFS'  
COUNTER #MARK7;
```

/** IDATA OFFSETS FOLLOW

CONSTANT

```
(  
"CLUSTER_SIZE",  
"ACTIVE_KEY",  
"ADDED_COUNT",  
"ADDED_COUNT_LOW",  
"ADDED_COUNT_HIGH",  
"BLOCKS_IN_BUCKET",  
"BUCKET_WEIGHT",  
"CARR_CTRL",  
"CONTROL_SIZE",  
"CURRENT_FUNCTION",  
"DESIGN_CYCLE",  
"DESIRED_FILL",  
"FILL_LOW",  
"FILL_HIGH",  
"GLOBAL_COUNT",  
"GRANULARITY",  
"INITIAL_COUNT",  
"INITIAL_COUNT_LOW",  
"INITIAL_COUNT_HIGH",  
"KEY_POSITION",  
"KEY_LOW",  
"KEY_HIGH",  
"KEY_SIZE",  
"KEY_TYPE",  
"LOAD_METHOD",  
"MAX_RECORD_SIZE",  
"MEAN_RECORD_SIZE",  
"NUMBER_DUPS",  
"NUMBER_KEYS",  
"NUMBER_RECORDS",  
"PROLOGUE_VERSION",  
"PROMPTING",  
"RECORD_FORMAT",  
"RESPONSES",  
"SCRIPT_OPTION",  
"SET_FUNCTION",  
"SIZE_LOW",  
"SIZE_HIGH",  
"SURFACE_OPTION",  
"TEST_PRIMARY",  
"TEST_SECONDARY",  
"TEST_SECONDARY_VALUE"
```

/** QUALIFIER TYPE

/** QUALIFIER TYPE

/** SYNONYM FOR INITIAL COUNT

/** QUALIFIER TYPE

```
) EQUALS #MARK7+1
INCREMENT 1
PREFIX "EDF$"
COUNTER #MARK8;
```

/** Array boundaries definitions

```
CONSTANT      "EDF$K_SDATASTART"    EQUALS #MARK3+1;
CONSTANT      "EDF$K_SDATAEND"      EQUALS #MARK5;

CONSTANT      "EDF$K_RDATASTART"    EQUALS #MARK2+1;
CONSTANT      "EDF$K_RDATAEND"      EQUALS #MARK6;

CONSTANT      "EDF$K_BDATASTART"    EQUALS #MARK1+1;
CONSTANT      "EDF$K_BDATAEND"      EQUALS #MARK7;

CONSTANT      "EDF$K_IDATASTART"    EQUALS #MARK0;
CONSTANT      "EDF$K_IDATAEND"      EQUALS #MARK8;

CONSTANT      "EDF$K_VDATASTART"    EQUALS #MARK4+1;
CONSTANT      "EDF$K_VDATAEND"      EQUALS #MARK8;

CONSTANT      "EDF$K_QTABSTART"     EQUALS #MARK4+1;
CONSTANT      "EDF$K_QTABEND"       EQUALS #MARK8;
```

/** Insert_object collision-action definitions

```
CONSTANT      (
    "REPLACE_OBJ",
    "IGNORE_OBJ",
    "AFTER_OBJ"

    ) EQUALS 0
INCREMENT 1
PREFIX "I";
```

/** Where definitions

```
CONSTANT      (
    "IF_FULL_PROMPT",
    "LOWER_AREA",
    "PAUSE",
    "SCREEN"

    ) EQUALS 0
INCREMENT 1
PREFIX "W";
```

/** Bucket-weight definitions

```
CONSTANT      (
    "SMALLER_BUFFERS",
```

```
"FLATTER_FILES"
```

```
) EQUALS 0  
INCREMENT 1  
PREFIX "EDFS";
```

```
/** Load-method definitions
```

```
CONSTANT (  
    "FAST_CONVERT",  
    "NOFAST_CONVERT",  
    "RMS_PUTS"  
  
    ) EQUALS 0  
    INCREMENT 1  
    PREFIX "EDFS";
```

```
/** Surface-option definitions
```

```
CONSTANT (  
    "FILL_SURFACE",  
    "SIZE_SURFACE",  
    "INIT_SURFACE",  
    "ADDED_SURFACE",  
    "KEY_SURFACE",  
    "LINE_SURFACE"  
  
    ) EQUALS 0  
    INCREMENT 1  
    PREFIX "EDFS";
```

```
/** Current-function definitions
```

```
CONSTANT (  
    "ADD",  
    "DELETE",  
    "EXIT",  
    "HELP",  
    "INVOKED",  
    "MODIFY",  
    "QUIT",  
    "SET",  
    "VIEW"  
  
    ) EQUALS 0  
    INCREMENT 1  
    PREFIX "EDFS";
```

```
/** Set_function definition
```

```
CONSTANT (  
    "SET_ANALYSIS",
```



```
"SET_DISPLAY"  
"SET_EMPHASIS"  
"SET_GRANULARITY"  
"SET_NUMBER_KEYS"  
"SET_OUTPUT"  
"SET_PROMPTING"  
"SET_RESPONSES"  
"SET_SCRIPT"  
"SET_TEMPLATÉ"
```

```
) EQUALS 0  
INCREMENT 1  
PREFIX 'EDFS';
```

/** Responses definitions

```
CONSTANT (  
    "AUTO",  
    "MAN"  
  
    ) EQUALS 0  
    INCREMENT 1  
    PREFIX 'EDFS';
```

/** Prompting definitions

```
CONSTANT (  
    "FULL",  
    "BRIEF"  
  
    ) EQUALS 0  
    INCREMENT 1  
    PREFIX 'EDFS';
```

/** Granularity definitions

```
CONSTANT (  
    "ONE",  
    "TWO",  
    "THREE",  
    "FOUR",  
    "DOUBLÉ"  
  
    ) EQUALS 0  
    INCREMENT 1  
    PREFIX 'EDFS';
```

/** Script-option definitions

```
CONSTANT (  
    "ADD_KEY_FDL",  
    "DELETE_KEY_FDL",
```

```
"IDX_DESIGN_FDL",
"REL_DESIGN_FDL",
"SEQ_DESIGN_FDL",
"OPTIMIZE_FDL",
"REDESIGN_FDL",
"ZERO_SCRIPT"
```

```
) EQUALS 0
INCREMENT 1
PREFIX "EDF$";
```

```
/** Yes-No definitions
```

```
CONSTANT (
    "NO",
    "YES",
    ) EQUALS 0
    INCREMENT 1
    PREFIX "EDF$";
```

```
/** Answer-class definitions
```

```
CONSTANT (
    "STRING_ANSWER",
    "REAL_ANSWER",
    "BOOLEAN_ANSWER",
    "INTEGER_ANSWER",
    "KEYWORD_ANSWER",
    "NO_ANSWER",
    "OBJECT_ANSWER",
    ) EQUALS 0
    INCREMENT 1
    PREFIX "" TAG "";
```

```
/** Equivalences for Design-cycle definitions
```

CONSTANT	"EDFSK_RS"	EQUALS "EDFSK_MEAN_RECORD_SIZE";
CONSTANT	"EDFSK_KL"	EQUALS "EDFSK_KEY_SIZE";
CONSTANT	"EDFSK_KP"	EQUALS "EDFSK_KEY_POSITION";
CONSTANT	"EDFSK_IL"	EQUALS "EDFSK_INITIAL_COUNT";
CONSTANT	"EDFSK_BF"	EQUALS "EDFSK_DESIRED_FILL";
CONSTANT	"EDFSK_EM"	EQUALS "EDFSK_BUCKET_WEIGHT";
CONSTANT	"EDFSK_RF"	EQUALS "EDFSK_RECORD_FORMAT";
CONSTANT	"EDFSK_RC"	EQUALS "EDFSK_DATA_RECORD_COMP";
CONSTANT	"EDFSK_KC"	EQUALS "EDFSK_DATA_KEY_COMP";
CONSTANT	"EDFSK_DK"	EQUALS "EDFSK_KEY_DUPS";
CONSTANT	"EDFSK_LM"	EQUALS "EDFSK_LOAD_METHOD";
CONSTANT	"EDFSK_AR"	EQUALS "EDFSK_ADDED_COUNT";
CONSTANT	"EDFSK_IC"	EQUALS "EDFSK_INDEX_RECORD_COMP";
CONSTANT	"EDFSK_WP"	EQUALS "EDFSK_WRITE_PLOT";
CONSTANT	"EDFSK_FINIS"	EQUALS "EDFSK_FINAL_DESIGN";
CONSTANT	"EDFSK_PV"	EQUALS "EDFSK_PROLOGUE_VERSION";

```
CONSTANT      'EDFSK_KT'          EQUALS 'EDFSK_KEY_TYPE';

/*  The following are used to interface to the EDF$GRAPH routine.
/*
  CONSTANT (
    LINE
    , SRF_INCREASING
    , SRF_DECREASING
    ) EQUALS 0 INCREMENT 1 PREFIX 'EDF$' TAG 'C';

  CONSTANT (
    BACKGROUND_COLOR
    , DARK_RED
    , MEDIUM_YELLOW
    , LIGHT_GREEN
    ) EQUALS 0 INCREMENT 1 PREFIX 'EDF$' TAG 'C';

END_MODULE EDFSTRUCT;
```


0124 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY